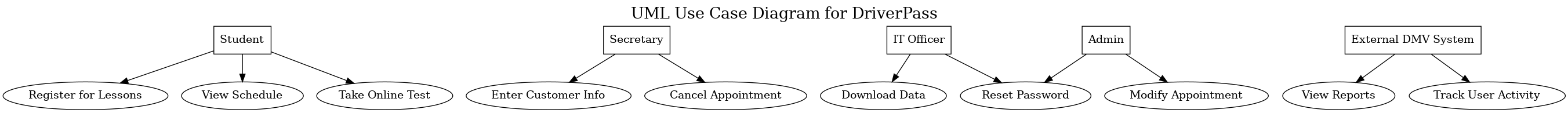
# CS 255 System Design Document Template

This template lays out all the different sections that you need to complete for Project Two. Each section has guidance to prompt your thinking. You will need to continually reference the interview transcript as you work to make sure that you are addressing your client’s needs. There is no required length for the final document. Instead the goal is to complete each section based on what your client’s needs are. Remove this note when you are finished, and replace all bracketed text with the relevant information.

## UML Diagrams

### UML Use Case Diagram

*[In Module Six, you were asked to complete a use case diagram based on your system design. If you would like to make any adjustments to your diagram, please do so. Please insert your use case diagram here. Check to make sure that you included appropriate components and symbols and that your design meets the client’s needs.]*

* *Use Case Actors: Student, Secretary, Admin, IT Officer, External DMV System*
* *Main Use Cases: Register for Lessons, View Schedule, Take Online Test, Enter Customer Info, Cancel Appointment, Reset Password, Download Data, Modify Appointment, View Reports, Track User Activity*

### UML Activity Diagrams

*[You were asked to choose* ***two*** *use cases and create* ***two*** *activity diagrams, one for each use case. Please insert* ***both*** *of your activity diagrams here. Check to make sure that you included appropriate components and symbols and that your design meets the client’s needs.]*

A diagram of a student login

AI-generated content may be incorrect.

1. **Register for Lessons**

* Student logs into system
* Selects driving package
* Picks lesson date and time
* Confirms reservation
* Process ends

A flowchart of a computer program

AI-generated content may be incorrect.

1. **Reset Password**

* Student clicks "Forgot Password"
* Enters email address
* System verifies email
* System sends password reset link
* Student enters new password
* Process ends

### UML Sequence Diagram

*[You were asked to create a sequence diagram based on* ***one*** *of the use cases you chose. Please insert your sequence diagram here. Check to make sure that you included appropriate components and symbols and that your design meets the client’s needs.]*

*A diagram of a register

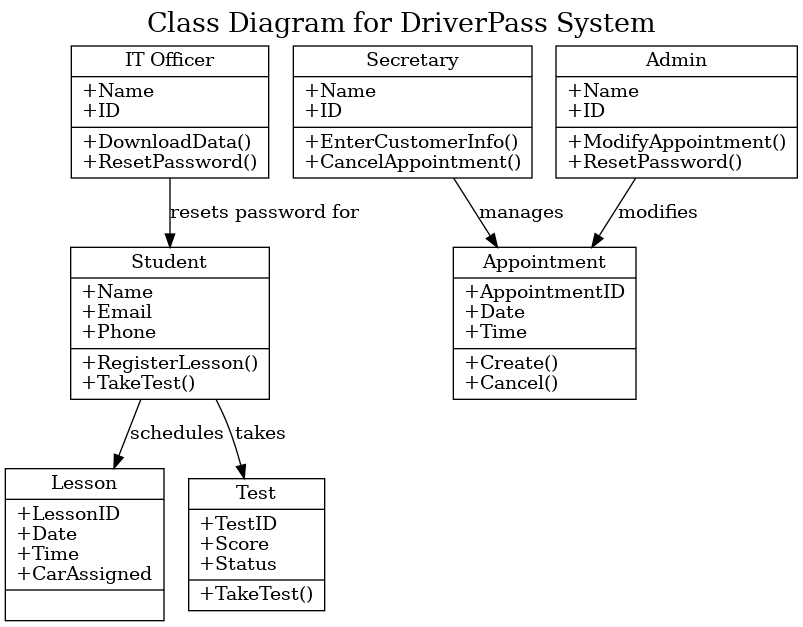
AI-generated content may be incorrect.*

**Register for Lessons Sequence**

* Student sends login request to System
* System returns login success
* Student selects a package
* Student requests available times from Scheduler
* Scheduler sends available times back
* Student chooses a date and time
* System saves reservation in Database
* Database confirms reservation
* System sends confirmation to Student

### UML Class Diagram

*[You were asked to create a class diagram based on the different classes and attributes needed for your system design. You are* ***not*** *required to include methods, but you may if you wish. Please insert your class diagram here. Check to make sure that you included appropriate components and symbols and that your design meets the client’s requirements.]*

**

Classes included:

* **Student** (Name, Email, Phone | RegisterLesson(), TakeTest())
* **Secretary** (Name, ID | EnterCustomerInfo(), CancelAppointment())
* **Admin** (Name, ID | ModifyAppointment(), ResetPassword())
* **IT Officer** (Name, ID | DownloadData(), ResetPassword())
* **Lesson** (LessonID, Date, Time, CarAssigned)
* **Appointment** (AppointmentID, Date, Time | Create(), Cancel())
* **Test** (TestID, Score, Status | TakeTest())

Relationships:

* Student schedules Lesson
* Student takes Test
* Secretary manages Appointment
* Admin modifies Appointment
* IT Officer resets Student password

## Technical Requirements

*[Based on the diagrams you have created, describe the technical requirements of your system. These requirements should address the required hardware, software, tools, and infrastructure necessary for your system design.]*

***Hardware Requirements***

* *Cloud server hosting (e.g., AWS, Azure, or Google Cloud)*
* *Client devices (laptops, desktops, mobile devices) with internet access*

***Software Requirements***

* *Web browsers (Google Chrome, Mozilla Firefox, Safari, Microsoft Edge)*
* *Backend system written in a web-friendly language (e.g., Python/Django, Java/Spring Boot)*
* *Frontend framework (e.g., React.js, Vue.js)*
* *Relational Database Management System (e.g., MySQL, PostgreSQL)*
* *CASE tool for diagram management (e.g., Lucidchart)*

***Tools and Development Infrastructure***

* *Secure web hosting with SSL/TLS encryption*
* *Regular database backups and redundancy setup*
* *User authentication tools (secure login, password recovery mechanisms)*
* *Access control system for different user roles (Student, Secretary, Admin, IT Officer)*
* *Automatic DMV data update interface (future integration)*

***Networking/Infrastructure Requirements***

* *Cloud platform ensures scalability and disaster recovery*
* *Minimal local storage requirements as most data resides in the cloud*
* *Secure VPN access for IT admin-level activities (optional)*